Future Generation Computer Systems 110 (2020) 610–619

Contents lists available at ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

CorClustST—Correlation-based clustering of big spatio-temporal datasets

Marc Hüsch [a,*], Bruno U. Schyska [b], Lueder von Bremen [c,b]

[a] TU Dortmund, Fakultät Statistik, 44221 Dortmund, Germany
[b] Carl von Ossietzky Universität Oldenburg, ForWind, 26129 Oldenburg, Germany
[c] DLR-Institut für Vernetzte Energiesysteme e.V., 26129 Oldenburg, Germany

HIGHLIGHTS

- A new spatio-temporal clustering algorithm is proposed.
- It utilizes spatial correlations over time to find meaningful clusters.
- The clustering strategy is effective for big data processing and data reduction.
- Its usefulness is demonstrated in a test case for wind power forecast errors.
- An extension of the algorithm allows for a large-scale parallelization.

ARTICLE INFO

Article history:
Received 23 June 2017
Received in revised form 9 February 2018
Accepted 1 April 2018
Available online 7 April 2018

Keywords:
Clustering
Big spatio-temporal data
Spatial dependence
Preprocessing
Data reduction

ABSTRACT

Increasing amounts of high-velocity spatio-temporal data reinforce the need for clustering algorithms which are effective for big data processing and data reduction. As currently applied spatio-temporal clustering algorithms have certain drawbacks regarding the comparability of the results, we propose an alternative spatio-temporal clustering technique which is based on empirical spatial correlations over time. As a key feature, CorClustST makes it easily possible to compare and interpret clustering results for different scenarios such as multiple underlying variables or varying time frames. In a test case, we show that the clustering strategy successfully identifies increasing spatial correlations of wind power forecast errors in Europe for longer forecast horizons. An extension of the clustering algorithm is finally presented which allows for a large-scale parallel implementation and helps to circumvent memory limitations. The proposed method will especially be helpful for researchers who aim to preprocess big spatio-temporal datasets and who intend to compare clustering results and spatial dependencies for different scenarios.

1. Introduction

Due to the continuing growth of data in many areas of application, there is an urgent need for efficient machine learning and data mining algorithms that can identify patterns in massive datasets. Clustering algorithms [1–3] are a popular tool to systematically detect groups of objects that share similar characteristics and can help to preprocess, reduce and compress big datasets. Especially in environmental applications, data is often available in high frequency across space and time which leads to special requirements for clustering methods to identify objects that are similar regarding both dimensions. The most popular clustering algorithms like the *k*-means algorithm [4] or hierarchical clustering algorithms [5–7] were, however, not developed specifically for spatio-temporal data and they do not take the special characteristics of such data into account. It is therefore important to develop novel spatio-temporal clustering algorithms which are able to efficiently extract information from big spatio-temporal datasets. In this sense, Kisilevich et al. (2010) [8] published a survey which gives an overview of common spatio-temporal clustering methods and applications. Most of the methods that are currently employed for clustering spatio-temporal data are based on density-based clustering techniques. DBSCAN [9], for instance, scans the entire data and marks each point as a core object (objects located in a cluster), border object (objects located at the border of a cluster), or noise (objects that are not located in a cluster) by determining the number of objects that are within a certain distance around each object. Birant and Kut (2007) [10] extended the idea of DBSCAN for spatio-temporal data. Their algorithm ST-DBSCAN marks each

* Corresponding author.
E-mail addresses: marc.huesch@tu-dortmund.de (M. Hüsch), bruno.schyska@uni-oldenburg.de (B.U. Schyska), lueder.von.bremen@uni-oldenburg.de (L. von Bremen).

https://doi.org/10.1016/j.future.2018.04.002
0167-739X/

object based on the number of objects that are within a certain spatial and a certain temporal distance. More recently, Agrawal et al. (2016) [11] developed a spatio-temporal clustering strategy which bases on the density-based clustering algorithm OPTICS (Ordering Points to Identify the Clustering Structure) [12]. Their proposed strategy ST-OPTICS first orders the observations in a dataset, then clusters them with a modified version of ST-DBSCAN and finally merges the resulting micro level clusters with an agglomerative algorithm. The authors showed in an application that the proposed strategy can lead to better clustering results than ST-DBSCAN in terms of cluster validity.

Despite considerable progress in the field of spatio-temporal clustering in recent years, currently applied methods still have some drawbacks that reinforce the need for alternative spatio-temporal clustering algorithms: Compared to the k-means algorithm, ST-DBSCAN and ST-OPTICS have the advantage that the number of clusters does not have to be predefined. One drawback of these algorithms is, however, that they do not directly find meaningful cluster centers which represent a certain cluster. This can be uncomfortable for a further analysis of cluster interconnections and especially for the purpose of big data reduction. In addition, the control parameters of these clustering algorithms need to be tuned in order to achieve an optimal clustering solution with respect to certain optimization criteria. In case of different control parameters, clustering results for different scenarios (e.g. varying time frames or multiple underlying variables) are, however, difficult to compare.

The considerations above have led to a novel approach for clustering big spatio-temporal datasets: By computing empirical correlations between pairs of spatial points over time, it is possible to find clusters that are easier to interpret, regardless of the number of time points and the underlying variable used in the analysis. As spatio-temporal data is often positively correlated only up to a certain spatial distance, it is not necessary to compute correlations for all pairs of points. By considering only those objects that are located within a certain spatial distance, the computation time and the memory requirements of the clustering technique are reduced drastically. CorClustST takes up some basic ideas of ST-DBSCAN and ST-OPTICS but provides a new concept in order to achieve a better comparability of clustering results for different scenarios. The applicability of the algorithm is exemplified with a cluster analysis of wind power forecast errors in Europe for different forecast horizons. It turns out that the algorithm successfully manages to represent increasing spatial correlations for longer forecast horizons by increased cluster sizes.

This paper is structured as follows: The different steps of the clustering strategy are presented in detail in Section 2. Subsequently, the algorithm is applied exemplarily for a clustering of wind power forecast errors in Section 3. As it is often not possible to cluster all observations from the entire spatial dimension due to computational reasons and memory limitations, Section 4 proposes an extension which makes it possible to combine clustering results from multiple subregions. The extension allows for a large-scale parallel implementation of the algorithm. In Section 5, the proposed algorithm is compared with several commonly used clustering methods regarding important features like complexity and interpretability. The ideas from this paper are finally reviewed in Section 6 and an outlook for future research is given.

2. Description of the clustering strategy

Definitions. The description of the clustering strategy is based on a set of N spatial points $D = \{s_1, s_2, \ldots, s_N\}$ and we assume that we observe attribute values $z(s_i, t)$ for all of these spatial points $s_i \in D$ for T different time steps $t = 1, \ldots, T$. Before the clustering strategy is described in detail, some necessary definitions are presented (compare [10]). Definition 1 first defines the general idea of a clustering process. An advantage of the proposed clustering algorithm is that not all of the points in the dataset have to be assigned to a cluster. Points that are not similar to other points in a spatio-temporal context will be declared as noise points.

Definition 1. The process of splitting D into c disjunct clusters $C_i \subseteq D, (i = 1, 2, \ldots, c)$, $\cup_{i=1}^{c} C_i \cup \{\text{NoisePoints}\} = D$, is called **clustering**.

The clustering algorithm proposed in this paper utilizes neighborhoods of spatial points to find such a clustering of points with similar characteristics. In a spatio-temporal context, two points can be similar over the spatial domain and the time domain. The spatial neighborhood of a spatial point, which is defined in Definition 2, contains points which are located close to the respective point in a spatial context.

Definition 2. The **spatial ϵ-neighborhood** of a spatial point s is defined as

$$\text{SpatNeigh}(s) = \{q \in D | \text{dist}(s, q) \leq \epsilon\},$$

where $\text{dist}(s, q)$ denotes a certain distance measure. Points that are located within the spatial ϵ-neighborhood are called **spatial neighbors** of s.

As we are not only interested in points that are located close to each other in a spatial context but also over the time domain, the empirical correlation of the spatial neighbors over time is taken into account in the clustering approach. Pearson's sample correlation [13] between the time series corresponding to two spatial points is used to define the spatio-temporal neighborhood in Definition 3. Rank correlation coefficients like empirical Spearman's rho [14] or empirical Kendall's tau [15] can be used alternatively.

Definition 3. The **spatio-temporal ρ-neighborhood** of a spatial point s is defined as

$$\text{SpatTempNeigh}(s) = \{q \in \text{SpatNeigh}(s) | \text{cor}(s, q) > \rho\},$$

where

$$\text{cor}(s, q) := \frac{\sum_{t=1}^{T}(z(s,t) - \bar{z}_s)(z(q,t) - \bar{z}_q)}{\sqrt{\sum_{t=1}^{T}(z(s,t) - \bar{z}_s)^2}\sqrt{\sum_{t=1}^{T}(z(q,t) - \bar{z}_q)^2}}$$

denotes Pearson's sample correlation coefficient over time between the spatial points s and q and $\bar{z}_s$ is the sample mean of the attribute values observed for the spatial point s. Points that are located within this neighborhood are called **spatio-temporal neighbors** of s.

With the definitions introduced above, it is now possible to provide a detailed description of the different steps of the clustering approach.

Clustering strategy. The first step of the clustering algorithm is the computationally most intensive part. Initially, sample correlations to all spatial neighbors that are located within a certain distance ϵ are computed for all spatial points. The distance ϵ has to be chosen in a smart way that the computation time is reasonable but also enough neighbors are considered. As all the pairwise correlations can be computed independently of each other, the computation can be performed easily in parallel by using multiple processing elements. This makes the algorithm performant even for big spatio-temporal datasets.

To conduct the further steps of the clustering approach, the number of spatial neighbors to which the correlation is greater or equal than a predefined value of ρ is initially determined for each spatial point. These points are called spatio-temporal neighbors as defined in Definition 3. Reasonable values for ρ can be found

in the literature [16]. For instance, a value of $\rho = 0.9$ could be used in a high correlation scenario and a value of $\rho = 0.7$ in a moderate correlation scenario. The spatial points are then arranged in descending order according to their number of spatio-temporal neighbors. This order is saved in a list $O = \{o_1, \ldots, o_N\}$.

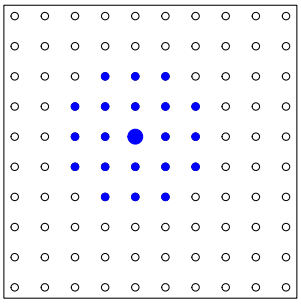
Step 1. *For all spatial points $s \in D$: Compute Pearson's sample correlation coefficient $cor(s, q)$ to all spatial neighbors $q \in SpatNeigh(s)$ for a predefined value of ϵ. Arrange the points in descending order according to the number of spatio-temporal neighbors $|SpatTempNeigh(s)|$ for a predefined value of ρ and save this order in a list $O = \{o_1, \ldots, o_N\}$.*

Subsequently, the spatial points are clustered based on their spatio-temporal neighbors. We include a simple running example (a grid of 10×10 spatial points) in the description to visualize the different steps of the clustering approach. Initially, the point with the highest number of spatio-temporal neighbors is chosen as the cluster center of cluster C_1 and all spatio-temporal neighbors are assigned to this cluster.

Step 2. *Choose the point*

$o_1 = argmax_{s \in D} |SpatTempNeigh(s)| \in O$

for a predefined value of ρ as a center point of the cluster C_1. All spatio-temporal neighbors $q \in SpatTempNeigh(o_1)$ are assigned to cluster C_1.



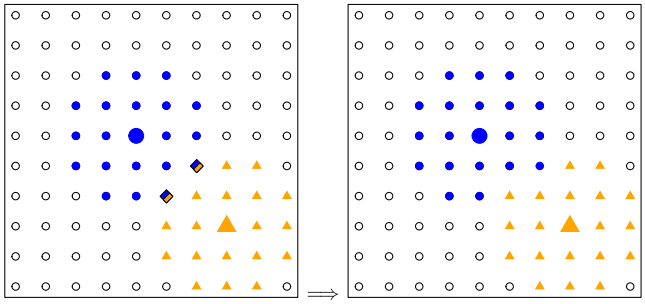
For all following ordered spatial points o_i in O, the following clustering procedure is then conducted iteratively: If o_i does not belong to a cluster and if more than 50% of the spatio-temporal neighbors of o_i do also not belong to a cluster, o_i is considered as a center point of a cluster. With the restriction that more than half of the spatio-temporal neighbors of o_i shall not belong to a cluster, it is guaranteed that points which are located close to the border of an existing cluster are not marked as a new cluster center. A markedly smaller threshold than 50% would lead to new cluster centers close to the border of already existing clusters. It would then be difficult to assign spatial points which belong to the spatio-temporal ρ-neighborhood of both the existing cluster center and the new cluster center to one of the two clusters. On the other hand, a higher threshold would result in many non-overlapping clusters which leads to the problem that many small clusters could arise in the areas close to the borders of two non-overlapping clusters. Therefore, the threshold value of 50% seems to be a good compromise in order to achieve a relatively smooth clustering result.

The current cluster label is then increased by one and o_i is assigned to the respective cluster. For all spatio-temporal neighbors of o_i it is checked subsequently whether they already belong to a cluster. If a neighbor does not yet belong to a cluster, it is assigned to the current cluster. If a neighbor already belongs to a cluster, it is checked whether the correlation of the neighbor to o_i is greater than the correlation of the neighbor to the center of its present cluster. If this fact is true, the cluster value of the neighbor is also changed to the current cluster label. Subsequently, the next point o_{i+1} is processed. The procedure is summarized in the clustering Steps 3 and 4.

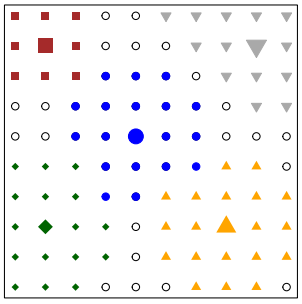
Step 3. *Choose the next point*

$o_2 = argmax_{s \in \{D \setminus o_1\}} |SpatTempNeigh(s)| \in O$

(with more than 50% of the neighbors not belonging to a cluster) as cluster center of cluster C_2. For all points $q \in SpatTempNeigh(o_2)$: Assign q to cluster C_2, if q does not yet belong to a cluster or if the correlation to o_2 is greater than the correlation to the current cluster center.

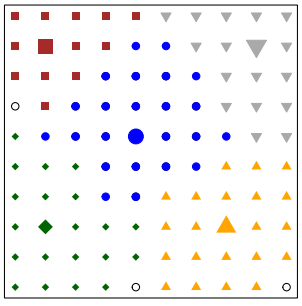


Step 4. *Repeat Step 3 for $o_3, \ldots, o_N$ until all points in O are processed.*



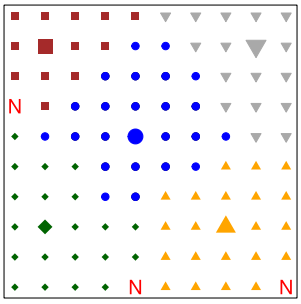
Due to the structure of this clustering strategy, it may occur that points are not assigned to a cluster although they have spatio-temporal neighbors (e.g. if more than half of the neighbors already belong to a cluster so that the point is not regarded as a cluster center and if the point is not in the ρ-neighborhood of one of the other cluster centers). As only those points shall be declared as noise points that are not similar to any other spatial point, it is required to assign these border points to the cluster to which the most similar spatio-temporal neighbor belongs. This is done in Step 5.

Step 5. *Points which have spatio-temporal neighbors but do not belong to a cluster after Step 4 are assigned to the cluster to which the most similar spatio-temporal neighbor (with the highest correlation to the respective point) belongs.*



For higher predefined values of ρ, it is likely that some of the spatial points will not have any spatio-temporal neighbors. These points are declared as noise points in Step 6 of the algorithm. This finalizes the clustering process.

Step 6. *Points of the dataset which are not assigned to a cluster C_i (with $i = 1, \ldots, c$) after Step 5 are finally declared as noise points.*



The proposed clustering approach allows to compare and interpret different clustering results depending on the predefined strength of correlation. Furthermore, the method makes it possible to identify spatial regions with higher and lower dependencies and can therefore be helpful to extract valuable information about the dependence structures in big spatio-temporal datasets. As an example, the algorithm is used in the following section to analyze the impact of the forecast horizon on spatial clustering of wind power forecast errors in Europe.

3. Test case: The impact of the forecast horizon on spatial clustering of wind power forecast errors in Europe

In times of increasing penetration rates of renewable energy, reliable forecasts of fluctuating energy sources such as wind power are getting more and more important. Load flow calculations for forecasted wind power in Europe need to be accurate, for example to predict transnational electricity flows or to provide backup capacities from reserve power plants. In the calculations, it is therefore necessary to consider errors in wind power forecasts. Regarding the influence of the forecast horizon on wind power forecasts, it is common knowledge that the quality of a forecast decreases the further one predicts into the future. For instance, it was shown in [17] for a test site located in Hilkenbrook (Germany) that the skill of wind speed and wind power forecasts decreases for longer forecast horizons up to 48 h. However, it has only barely been discussed that the spatial correlation of wind power forecast errors also increases for longer forecast horizons. This issue was first discussed in [18], where the authors state that growing systematic errors for increasing forecast horizons lead to higher spatial correlations. In a case study of western Denmark [19], it was demonstrated that wind power forecast errors are only slightly correlated in a spatial context for short forecast horizons. It can be expected that this effect increases when longer forecast horizons are considered. Especially in those regions where large forecast errors occur and a high amount of wind plants is installed, high spatial correlations could mean increasing risks for various stakeholders due to higher cumulative forecast errors.

In order to investigate this aspect, the proposed algorithm is used to characterize the influence of the forecast horizon and other possible influence factors on a spatial clustering of wind power forecast errors. The analysis is conducted for onshore regions across Europe over the period from April 2010 to February 2016. The wind power forecasts are generated from deterministic wind speed forecasts of the European Centre for Medium-Range Weather Forecasts (ECMWF) [20] of 100m height by using a regional onshore power curve proposed in [21]. Six different forecast horizons (12, 24, 36, 48, 60 and 72 h) are considered. Forecasts are issued twice a day for all considered forecast horizons. The study region comprises onshore regions of 32 European countries that have a distinct amount of installed wind power capacity (compare [22]). To select onshore regions only, a land–sea-mask is used, which is provided by ECMWF for the grid of the deterministic forecasts. Forecast errors for a certain forecast horizon q are computed by subtracting the 0-hour forecast (initialized q hours after the q-hour forecast is made) from the q-hour forecast for each grid cell.

Due to the relatively high resolution of the forecasts (horizontal grid spacing of approximately 16 km) and the considered study time, the forecast error datasets for each forecast horizon comprise in total $N = 49968$ spatial points and $T = 4316$ equidistant time steps. For such a large number of data points, it is computationally intensive to perform a clustering with common algorithms like hierarchical clustering algorithms. These mostly require to store large distance matrices with $\frac{N(N-1)}{2}$ entries. This can be hard to accomplish even with computers that have a large amount of available memory. The clustering approach proposed in this paper, however, avoids this problem by incorporating the fact that only points in a given spatial neighborhood will likely be correlated to each other. The clustering is performed for two degrees of correlation. In a moderate correlation scenario ρ is set to 0.7 and in a high correlation scenario ρ is set to 0.9. With these relatively high values of ρ, it is unlikely that clusters will be found that comprise grid cells that are located far away from each other. A preceding analysis showed that the data points tend to be uncorrelated for distances that exceed 600 km, regardless of the considered forecast horizon. Therefore, the value of ϵ for the clustering approach is set to 600 km. The cluster analysis is performed for all six considered forecast horizons from 12 h to 72 h for both correlation scenarios.

To get a first impression about the differences in the clustering results, the resulting numbers of clusters are compared for the different forecast horizons in Fig. 1. The comparison indicates that the forecast horizon has a substantial influence on spatial dependence of wind power forecast errors.

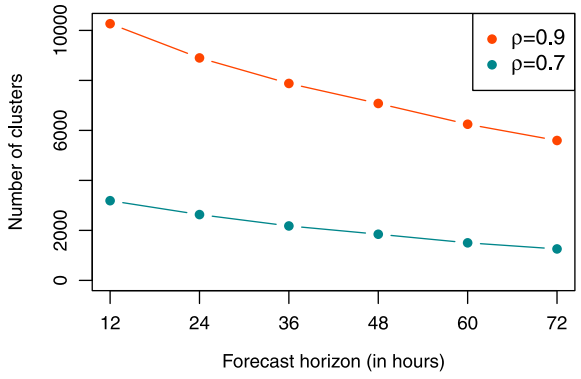
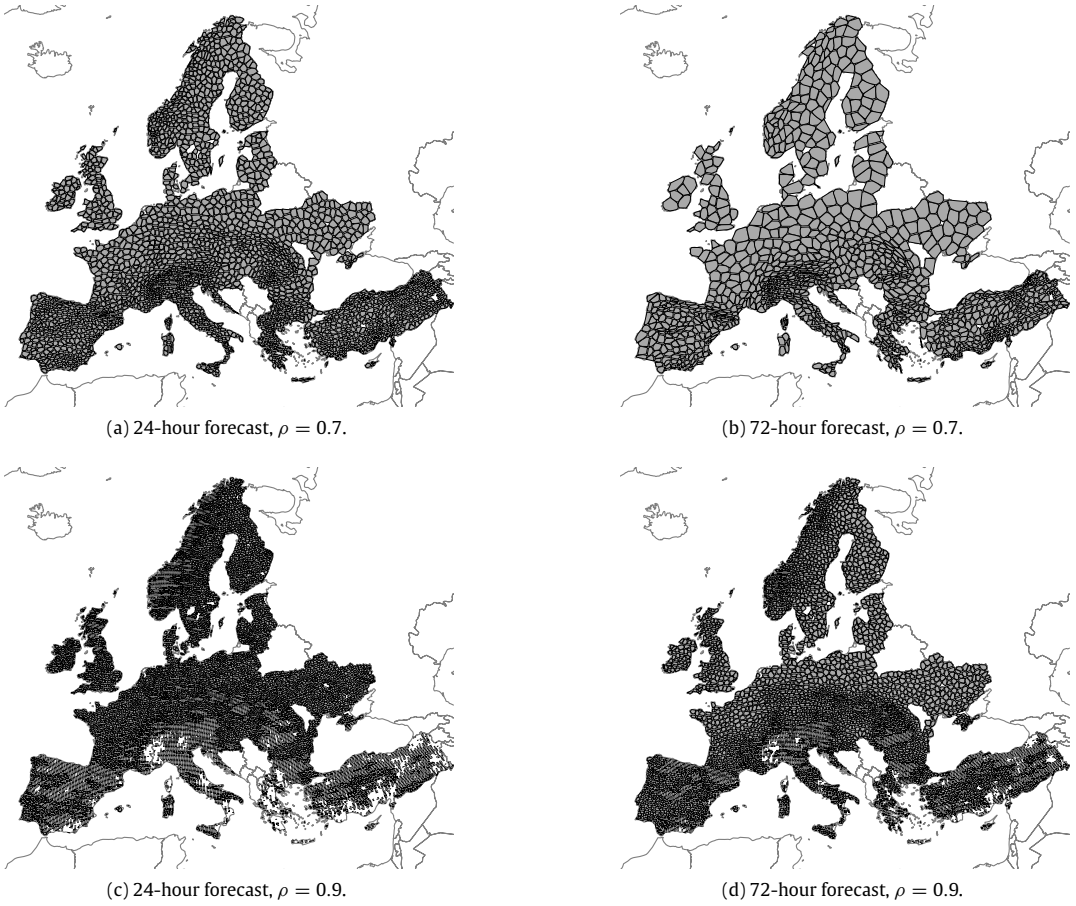


Fig. 1. Number of clusters for different forecast horizons.

In the moderate correlation scenario ($\rho = 0.7$), the number of clusters decreases from 3189 clusters for the 12-hour forecasts to 1259 clusters for the 72-hour forecasts. In the high correlation scenario, 10269 clusters are found for the 12-hour forecasts and 5597 clusters for the 72-hour forecasts. Therefore, the cluster analysis confirms that spatial dependence of wind power forecast errors increases substantially for longer forecast horizons as stated by [18].

In order to compare different regions of Europe according to their degree of spatial forecast error dependence, we focus on a comparison of the clustering results for the 24- and 72-hour



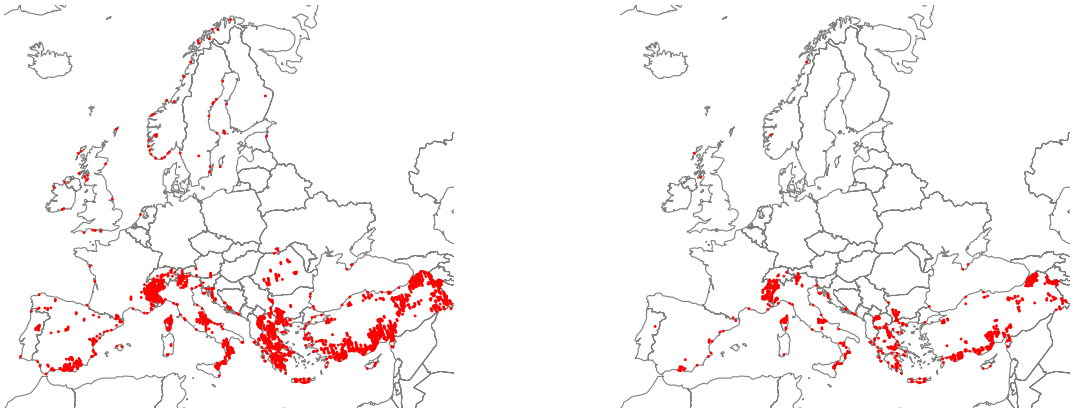
(a) 24-hour forecast, $\rho = 0.7$.

(b) 72-hour forecast, $\rho = 0.7$.

(c) 24-hour forecast, $\rho = 0.9$.

(d) 72-hour forecast, $\rho = 0.9$.

Fig. 2. Clustering of 24- and 72-hour forecast errors.



(a) 24-hour forecast.

(b) 72-hour forecast.

Fig. 3. Noise points found with the 24- and 72-hour forecast error clustering ($\rho = 0.9$).

forecast errors. In Fig. 2, the resulting clusters for the moderate ($\rho = 0.7$) and the high ($\rho = 0.9$) correlation scenario are visualized in a map of Europe.

Comparing the two forecast horizons, the clusters for the 72-hour forecast errors are markedly larger than the clusters for the 24-hour forecast errors in all parts of Europe in both correlation scenarios. Besides increasing cluster sizes for longer forecast horizons, Fig. 2 also reveals striking differences in the cluster sizes in different regions of Europe. In all scenarios, the largest clusters can be found in the low-terrain areas of Northern Europe, Germany, France and Eastern Europe. In these areas, the biggest increase in cluster sizes can be recognized when comparing the 24-hour and 72-hour forecast errors. This fact can be quite interesting since a relatively large amount of wind power plants is installed in these areas. A high correlation of wind power forecast errors of closely located wind power plants can lead to an increased cumulative forecast error for the respective area. The smallest clusters occur in mountain areas like the Alps, the Pyrenees and the Carpathian mountains and in the higher lying areas of Southern Europe and Norway.

With the proposed clustering approach, it is not only possible to explore areas with a relatively high spatial dependence, but also to identify the regions that are characterized by a high number of noise points. These grid cells do not have any spatial neighbor with a correlation higher than the predefined value of ρ. Areas with a high number of noise points can therefore be regarded as areas that are characterized by a low spatial correlation of wind power forecast errors. The noise points that are found for the high correlation scenario are visualized in Fig. 3 for the 24- and 72-hour forecast errors. For the 24-hour forecast errors, the majority of the noise points can be found in high mountain areas like the French Alps, the Apennines in Italy and the Taurus Mountains in Turkey. In addition, several noise points are located very close to the seaside. The points in coastal areas mostly disappear when regarding the 72-hour forecast horizon. However, a large number of noise points can still be found in the French Alps and in mountain areas of Greece and Turkey, whereas almost no noise points remain in Northern Europe. The number of noise points reduces markedly for longer forecast horizons which results from a general increase in spatial correlation.

4. Extension: Combination of clustering results from multiple subregions

The application in the previous section has demonstrated that the proposed algorithm can provide a highly efficient way to cluster big spatio-temporal datasets. Nevertheless, memory limitations can still be an issue when the number of spatial points N increases. Regarding the example of wind power forecast errors, this problem could occur, for instance, when other large countries like Russia shall also be considered in the analysis. In addition, the resolution of meteorological forecasts steadily increases in order to achieve a higher forecasting accuracy. Exemplarily, the European Centre for Medium-Range Weather Forecasts (ECMWF) reduced the horizontal grid spacing for the deterministic forecasts from 16 km to 9 km in March 2016 [23]. In order to prevent issues that may occur due to memory limitations, an extension of the clustering strategy is presented in this section. The extension builds on the idea that the full study region can be divided into multiple subregions for which a separate clustering can be conducted. As far distant points are generally not strongly related to each other, clusters located far away from another subregion should remain unaffected from splitting the study region. As an example, we conduct a clustering of 12-hour wind power forecast errors for spatial points located in Austria and Switzerland (for $\rho = 0.7$). The value of ϵ for the clustering approach is again set to 600 km and should therefore not affect the clustering results as the data points tend to be uncorrelated for long distances. The clustering is performed once for both countries jointly and once for the two countries individually. The resulting clusters for both approaches are visualized in Fig. 4. Clusters from the individual clustering that are different compared to the joint clustering are highlighted in red.

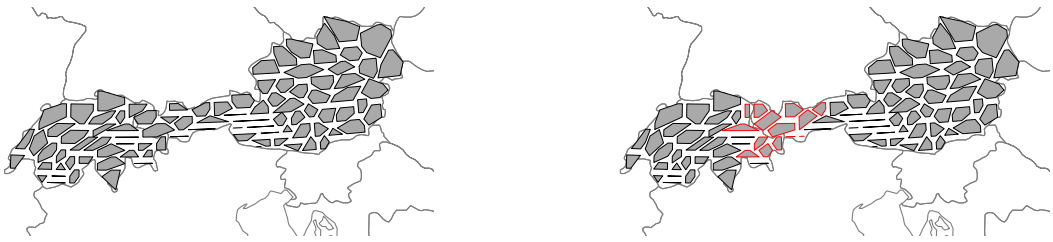
The comparison reveals that the clustering structure differs only in the region close to the border of the two countries. In order to combine the clustering results of multiple subregions, hence only those spatial points need to be reprocessed that belong to a cluster close to the border to another subregion. By defining inner edge clusters (clusters with at least one spatial point located within a distance δ_1 to the closest point of another subregion) and outer edge clusters (clusters not belonging to the inner edge clusters and with at least one spatial point located within a distance δ_2 to the closest point of another subregion), the spatial points in the border regions can be determined for which an additional clustering needs to be conducted. While the points located in the outer edge clusters keep their cluster labels (these clusters should still be similar to the clusters that are obtained with the clustering for the entire study region), the cluster labels of the spatial points belonging to the inner edge clusters are removed. For the example of clustering 12-hour wind power forecast errors in Austria and Switzerland (individual computation), the respective inner and outer edge clusters are visualized for two different choices of δ_1 and δ_2 in Fig. 5. A new dataset is created that contains the time series and the cluster labels of the spatial points belonging to one of the inner or one of the outer edge clusters and an additional clustering is performed for these points.

In order to present the idea of the clustering extension in detail, we assume that the full dataset D can be divided into R subdatasets (subregions) $D^{(1)}, \ldots, D^{(R)}$. The number of time steps T is required to be equal in each subdataset $D^{(r)} (r = 1, \ldots, R)$, whereas the number of spatial points $N^{(r)}$ may differ. For each subdataset, a separate clustering is initially performed with the algorithm proposed in Section 2. This results in a number of $c^{(r)}$ clusters for the respective subdataset $D^{(r)}$. Before the clusters from the different subdatasets can be combined, their cluster labels need to be changed in order to avoid clusters with the same label from different subdatasets. Therefore, the clusters from the first subdataset $D^{(1)}$ keep their cluster labels $C_1, \ldots, C_{c^{(1)}}$ whereas the clusters from the second subdataset $D^{(2)}$ receive the labels $C_{c^{(1)}+1}, \ldots, C_{c^{(1)}+c^{(2)}}$, the clusters from $D^{(3)}$ the labels $C_{c^{(1)}+c^{(2)}+1}, \ldots, C_{c^{(1)}+c^{(2)}+c^{(3)}}$ and so forth. Subsequently, the following steps can be performed to receive a clustering solution for the entire study region:

Step 1. Determine the set of inner edge points

$$s_{inner}^{(r)} = \{s \in D^{(r)} | \min_{q \in \{D^{(1)}, \ldots, D^{(R)}\} \setminus D^{(r)}} \text{dist}(s, q) \leq \delta_1\}$$

for each subdataset $D^{(1)}, \ldots, D^{(R)}$. Save the entire list of inner edge points in a set $s_{inner} = \cup_{r \in \{1, \ldots, R\}} s_{inner}^{(r)}$ and the corresponding cluster labels (uniquely) in a set C_{inner}.



(a) Computation jointly for both countries.

(b) Computation individually for each country; clusters that differ from the joint computation are highlighted in red.

Fig. 4. 12-hour forecast error clustering of spatial points located in Austria and Switzerland ($\rho = 0.7$). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

(a) $\delta_1 = 25$ km, $\delta_2 = 75$ km. (b) $\delta_1 = 100$ km, $\delta_2 = 150$ km.

Fig. 5. Edge clusters for the 12-hour forecast error clustering ($\rho = 0.7$) in Austria and Switzerland (individual computation) for two different choices of the parameters δ_1 and δ_2. Inner edge clusters are highlighted in dark blue and outer edge clusters in light blue. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Step 2. Determine the set of outer edge points

$$s_{outer}^{(r)} = \{s \in D^{(r)} | \min_{q \in \{D^{(1)},\dots,D^{(R)}\} \setminus D^{(r)}} \delta_1 < \text{dist}(s, q) \leq \delta_2\}$$

for each subdataset $D^{(1)}, \dots, D^{(R)}$. Save the entire list of outer edge points in a set $s_{outer} = \cup_{r \in \{1,\dots,R\}} s_{outer}^{(r)}$ and the corresponding cluster labels (uniquely) in a set C_{outer} (without clusters that already belong to C_{inner}).

Step 3. Create a new dataset D_{edge} with the spatial points that belong to a cluster listed in C_{inner} or C_{outer} and the points in s_{inner} that were marked as noise by the clustering algorithm.

Step 4. Set all cluster labels of the spatial points in D_{edge} that belong to a cluster listed in C_{inner} to zero and apply the clustering strategy proposed in Section 2 on the new dataset. This leads to new clusters that are marked by new cluster labels.

Step 5. Combine the clustering results for the border area from Step 4 with the results for the remaining spatial points. This leads to a final clustering solution.

In order to test the proposed approach, we consider again the example of clustering 12-hour wind power forecast errors in Austria and Switzerland. The clustering results that were obtained individually for each country are now combined with the clustering extension described above. The combination of the clustering results is performed once with comparatively small values of $\delta_1 = 25$ km and $\delta_2 = 75$ km and once with higher values of $\delta_1 = 100$ km and $\delta_2 = 150$ km. The results are visualized in Fig. 6.

Regarding the clusters that are obtained for small values of the parameters δ_1 and δ_2, differences compared to the joint computation are still prominent in the area close to the border. However, when the values of the parameters are increased, the clustering results tend to be more similar to those obtained with the joint computation. For values of $\delta_1 = 100$ km and $\delta_2 = 150$ km, it turns out that the clustering structure already equals the one observed with the joint computation. In general, both computation methods are able to highlight the same regions in which higher or lower spatial correlations are present and are thus able to capture the dependence structure equally well. The distances δ_1 and δ_2 need to be predefined related to the degree of correlation ρ that is chosen for the clustering process. As a higher value of ρ leads to smaller clusters and therefore to a more similar clustering structure close to the borders, generally smaller values of δ_1 and δ_2 may be selected than for small correlation thresholds. This makes the proposed method highly efficient and therefore attractive for many practitioners who intend to preprocess and reduce big spatio-temporal datasets. With the extension, the clustering process can easily be parallelized which leads to major improvements in performance.

5. Discussion: Comparison with other clustering algorithms

In order to distinctly point out the advantages of CorClustST and to discuss possible disadvantages, this section addresses differences and similarities between the proposed method and the most popular clustering algorithms that are currently employed for spatio-temporal data. Table 1 compares the proposed algorithm with several clustering methods regarding important features like complexity, memory requirements, parallelization and interpretability.

For big spatio-temporal datasets, the most popular clustering algorithms like the k-means algorithm or hierarchical clustering methods are generally less efficient than specifically designed spatio-temporal clustering methods like ST-DBSCAN or ST-OPTICS. Regarding the k-means algorithm, it is particularly necessary to perform the clustering for different values of k with heuristic criteria like the elbow criterion [26] to find an optimal clustering solution. This can be a major drawback in case of extremely large datasets. A hybrid clustering framework, however, could serve as an efficient alternative: Schyska et al. (2017) [27] propose a combination of the k-means algorithm and hierarchical clustering for the reference site selection of wind farms. They first cluster the

(a) $\delta_1 = 25$ km, $\delta_2 = 75$ km. (b) $\delta_1 = 100$ km, $\delta_2 = 150$ km.

Fig. 6. 12-hour forecast error clustering of spatial points located in Austria and Switzerland ($\rho = 0.7$). The clustering was conducted separately for each country and the clusters were combined subsequently with the proposed clustering extension. Clusters that differ from the joint computation are highlighted in red. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 1
Comparison of CorClustST with existing algorithms regarding complexity, parallelization and interpretability for spatio-temporal data.

Algorithm	Complexity, parallelization	Computation, interpretability for spatio-temporal datasets
k-means	• Complexity: $\mathcal{O}(N \cdot T \cdot k \cdot i)$, with k the number of clusters, T the number of time points and i the number of iterations [24] • Large-scale parallelization not directly possible	Disadvantages: • Not specifically designed for spatio-temporal data • The number of clusters has to be predefined with heuristic criteria • All observations have to be assigned to a cluster • Comparison of clustering results for different scenarios can be difficult Advantage: • Meaningful cluster centers are provided for the purpose of data reduction and for analyzing cluster interconnections
Hierarchical clustering (complete linkage)	• Complexity: $\mathcal{O}(N^2 \cdot \log N)$ [25] • Large-scale parallelization not directly possible, requires computation of a distance matrix with $\frac{N(N-1)}{2}$ entries	Disadvantages: • Not specifically designed for spatio-temporal data • Cluster centers are not directly provided • All observations have to be assigned to a cluster • Comparison of clustering results for different scenarios can be difficult Advantage: • The number of clusters does not have to be predefined (however a suitable stopping criterion needs to be chosen)
ST-DBSCAN, ST-OPTICS	• Complexity: $\mathcal{O}(N \cdot \log N)$ [11] • Large-scale parallelization difficult, could be achieved with a similar technique as proposed in Section 4	Disadvantages: • Cluster centers are not directly provided • Comparison of clustering results for different scenarios can be difficult Advantages: • Specifically designed for spatio-temporal data (high efficiency) • The number of clusters does not have to be predefined • Not all observations have to be assigned to a cluster, unusual observations are declared as noise points
CorClustST	• Complexity: $\mathcal{O}(N \cdot \log N)$ for small values of ϵ, $\mathcal{O}(N^2)$ for $\epsilon \to \infty$ • Large-scale parallelization possible with the extension proposed in Section 4	Disadvantages: • The clustering solution is not optimized regarding a specific quality criterion • Higher complexity than ST-DBSCAN and ST-OPTICS for large values of ϵ Advantages: • Specifically designed for spatio-temporal data (high efficiency) • The number of clusters does not have to be predefined • Not all observations have to be assigned to a cluster, unusual observations are declared as noise points • Meaningful cluster centers are provided for the purpose of data reduction and for analyzing cluster interconnections • Clusters for different scenarios and different spatial regions can be compared easily as cluster sizes depend directly on the degree of spatial correlation

locations into geographical clusters with the k-means algorithm and select the site with the highest wind power capacity for each cluster. Subsequently, pairwise (temporal) correlations between the selected sites are computed and a hierarchical clustering is applied to find the final reference sites. However, this approach is only valid when meaningful additional information (such as installed wind power capacity) is available.

CorClustST provides an efficient way to cluster all spatial points directly without the requirement of additional information: The proposed algorithm has a comparatively low complexity and requires only little memory space, especially when a small value is chosen for the parameter ϵ which controls the number of spatial neighbors considered for the clustering. For small values of ϵ, less spatial neighbors are taken into account and therefore in total less pairs of spatial points need to be processed. The complexity of CorClustST is then comparable to the complexity of ST-DBSCAN and ST-OPTICS. Contrary to these algorithms, CorClustST computes empirical correlations between predetermined pairs of spatial points before assigning them into clusters. This avoids using a stack that leads to multiple computations for the same pairs of points. By considering only relevant pairs of spatial points and by computing their correlations in advance, the clustering process is more easy to parallelize than the stacked versions of ST-DBSCAN and ST-OPTICS. Furthermore, the extension proposed in Section 4 allows for a large-scale parallel implementation that can markedly improve computation times. Existing spatio-temporal clustering methods do currently not allow for such a large-scale parallelization. Summarizing, the relatively short computation times of the proposed method allow to cluster even big spatio-temporal datasets in a short amount of time. This is demonstrated in Table 2, where we compare the execution times of the algorithm (non-parallel implementation) for the 24-hour wind power forecast error dataset for different values of ϵ and different numbers of spatial points N. The threshold ρ is exemplarily set to 0.9. As the complexity of the algorithm is mainly influenced by the number of spatial neighbors which is controlled only by ϵ, different values for the threshold ρ do not have a considerable influence on the computation times. Even for a relatively high value of $\epsilon = 600$ km, the comparison in Table 2 shows that the computation time increases only moderately for a large number of spatial points N. By computing the pairwise

Table 2
Computation time in seconds of CorClustST (non-parallel implementation) for different values of ϵ and N for the 24-hour forecast error dataset (for $\rho = 0.9$). The computation was performed in R [28] on a personal computer with an Intel Core i5-4460 CPU and 8GB RAM.

	$N = 1000$	$N = 10000$	$N = 20000$	$N = 49968$
$\epsilon = 100$ km	66.69	875.53	1438.81	3684.51
$\epsilon = 300$ km	208.59	5709.20	8981.60	21422.19
$\epsilon = 600$ km	236.71	16094.08	29004.67	74527.64

correlations in Step 1 of the algorithm in parallel, the computation times can be further reduced which then allows to cluster even big spatio-temporal datasets in a reasonable amount of time.

Regarding the interpretability of the results, the proposed clustering strategy combines different advantages of existing clustering techniques like the k-means algorithm and ST-DBSCAN: One very helpful feature of the k-means algorithm is that it provides cluster centers that can be utilized for the purpose of data reduction and for further analysis of cluster interconnections. However, the number of clusters k needs to be predetermined in advance which is a major drawback for big spatio-temporal datasets. The spatio-temporal clustering algorithms ST-DBSCAN and ST-OPTICS do not require a predefined number of clusters but they do, however, not provide meaningful cluster centers. These drawbacks are addressed by CorClustST. Here, the number of clusters does not have to be predefined in advance and meaningful cluster centers are provided which correspond to those spatial points with the highest number of spatio-temporal neighbors in certain regions. If a high value of ρ is chosen for determining the spatio-temporal neighbors, data reduction can be performed intuitively by focusing on the cluster centers that are characteristic for the respective regions. Another advantage of CorClustST compared to the k-means algorithm or hierarchical clustering methods is that it shares the possibility of density-based clustering algorithms to determine noise points.

For several applications such as the example of wind power forecast errors, it may be required that the applied clustering technique directly allows to compare the strength of possible spatial dependencies for different scenarios (e.g. different forecast horizons) and for different regions. In this sense, CorClustST clearly separates from the other discussed clustering strategies. The goal of common clustering algorithms mainly is to find an optimal clustering solution by minimizing the distances (defined by metrics like the Euclidean distance) between observations within a cluster and maximizing the distances between observations that belong to different clusters. This requires to optimize control parameters such as the number of clusters k for the k-means algorithm [4] or the parameters *MinPts*, *Eps*1, *Eps*2 and $\Delta\epsilon$ for ST-DBSCAN [10] in advance via heuristic criteria. As these parameters need to be adjusted to find optimal clustering solutions for different scenarios, the clustering results are difficult to compare because different control parameters lead to a completely different interpretation of the resulting clusters. Since CorClustST uses empirical correlations to determine the clusters, the algorithm allows to compare different clustering results by fixing the value of ρ (the desired degree of spatial correlation which is easy to interpret) for all scenarios.

Contrary to the other algorithms, the main goal of CorClustST is therefore not to find an optimal clustering solution regarding the (dis)similarity of the objects, but rather to provide an efficient descriptive tool to compare the degree of spatial dependence for different scenarios and different spatial regions. As CorClustST does not compete with the other discussed algorithms in this sense, we refrain from comparing the algorithms regarding cluster validity. Although CorClustST was not mainly designed for this purpose, the algorithm can still be a helpful tool when an optimal clustering solution shall be found: If the number of spatial points in the dataset is too large to perform a clustering efficiently with traditional clustering methods, CorClustST can first be applied with rather high correlation thresholds to reduce the dataset. The reduced dataset, which should consist of the cluster centers and the noise points that do not belong to a cluster, can subsequently be processed with the desired clustering technique in order to find an optimal clustering solution with respect to a specific quality criterion.

6. Conclusion and future work

Spatio-temporal clustering is a popular way to identify patterns in massive spatio-temporal datasets. As currently employed clustering methods still have some drawbacks regarding the comparability and the interpretability of the results, an alternative strategy for clustering big spatio-temporal datasets has been proposed in this paper. CorClustST clusters the spatial points in a dataset based on spatial correlations over time and makes it better possible to compare clustering results for varying periods of time and multiple underlying variables than with existing algorithms. In a test case, the algorithm successfully identified increasing spatial correlations of wind power forecast errors for longer forecast horizons and highlighted those regions of Europe in which spatial dependence is mostly prominent. It was also shown that the clustering method can be easily extended in such way that it allows for an efficient large-scale parallelization while preserving the essential clustering structure. With the proposed approach, a clustering of big spatio-temporal datasets can be performed even on systems with only little memory capacity. Other than currently employed methods, the clustering strategy additionally provides meaningful cluster centers which makes it especially valuable for the purposes of preprocessing and data reduction.

For future research, the insights gained with the clustering of wind power forecast errors in Section 3 increase the need for analyzing spatial dependence of wind power forecast errors in more detail. Spatio-temporal copulas [29], for instance, could be used to model the full dependence structure of wind power forecast errors over space and time and could allow to check whether longer forecast horizons also lead to increasing tail dependencies (i.e. whether extremely large forecast errors tend to occur jointly at closely located grid cells). By using calibrated meteorological ensemble forecasts [30–32], it could furthermore be possible to better assess the risks that occur due to spatial dependence of wind power forecasts for long forecast horizons. The information from the ensemble forecasts could, for instance, be used for grid security calculations and could also help to improve probabilistic electricity price forecasts [33,34].

Acknowledgments

We wish to thank Uwe Ligges from TU Dortmund University for helpful comments and suggestions. In addition, we would also like to thank the European Centre for Medium Range Weather Forecasts (ECMWF) for providing the wind speed data for the analysis. Furthermore, we kindly thank two referees for their constructive comments and very helpful suggestions.

During his research stay at the Center for Wind Energy Research (ForWind) in Oldenburg, Germany, Marc Hüsch has received funding from the Alumni-Verein Dortmunder Statistikerinnen und Statistiker, Germany. Bruno U. Schyska has received funding from the European Union's Seventh Programme for research, technological development and demonstration under grant agreement No. 609795 (IRPWind). Lueder von Bremen has received funding from the ministry of science and culture of Lower Saxony, Germany in the project ventus efficiens (No. ZN3024).

References

[1] L. Kaufman, P.J. Rousseeuw, Finding groups in data: An introduction to cluster analysis, in: Wiley Series in Probability and Statistics, Wiley, 2008.
[2] B. Everitt, S. Landau, M. Leese, D. Stahl, Cluster analysis, in: Wiley Series in Probability and Statistics, Wiley, 2011.
[3] K. Ericson, S. Pallickara, On the performance of high dimensional data clustering and classification algorithms, Future Gener. Comput. Syst. 29 (4) (2013) 1024–1034, Special Section: Utility and Cloud Computing.
[4] J.A. Hartigan, M.A. Wong, Algorithm AS 136: A K-means clustering algorithm, J. R. Stat. Soc. Ser. C. Appl. Stat. 28 (1) (1979) 100–108.

[5] R. Sibson, SLINK: An optimally efficient algorithm for the single-link cluster method, Comput. J. 16 (1) (1973) 30–34.
[6] D. Defays, An efficient algorithm for a complete link method, Comput. J. 20 (4) (1977) 364–366.
[7] J.H. Ward, Hierarchical grouping to optimize an objective function, J. Amer. Statist. Assoc. 58 (301) (1963) 236–244.
[8] S. Kisilevich, F. Mansmann, M. Nanni, S. Rinzivillo, Spatio-temporal clustering, in: O. Maimon, L. Rokach (Eds.), Data Mining and Knowledge Discovery Handbook, Springer US, 2010, pp. 855–874 Ch. 44.
[9] M. Ester, H.-P. Kriegel, J. Sander, X. Xu, A density-based algorithm for discovering clusters in large spatial databases with noise, in: KDD-96 Proceedings, AAAI Press, 1996, pp. 226–231.
[10] D. Birant, A. Kut, ST-DBSCAN: An algorithm for clustering spatial–temporal data, Data Knowl. Eng. 60 (1) (2007) 208–221.
[11] K. Agrawal, S. Garg, S. Sharma, P. Patel, Development and validation of OPTICS based spatio-temporal clustering technique, Inform. Sci. 369 (2016) 388–401.
[12] M. Ankerst, M.M. Breunig, H.-P. Kriegel, J. Sander, OPTICS: Ordering points to identify the clustering structure, in: Proceedings of the 1999 ACM SIGMOD International Conference on Management of Data, SIGMOD '99, ACM, 1999, pp. 49–60.
[13] K. Pearson, Note on regression and inheritance in the case of two parents, Proc. R. Soc. I 58 (1895) 240–242.
[14] C. Spearman, The proof and measurement of association between two things, Am. J. Psychol. 15 (1) (1904) 72–101.
[15] M.G. Kendall, A new measure of rank correlation, Biometrika 30 (1–2) (1938) 81.
[16] R. Taylor, Interpretation of the correlation coefficient: A basic review, J. Diagn. Med. Sonog. 6 (1) (1990) 35–39.
[17] M. Lange, On the uncertainty of wind power predictions - Analysis of the forecast accuracy and statistical distribution of errors, J. Sol. Energy Eng. 127 (2) (2005) 177–184.
[18] U. Focken, M. Lange, K. Mönnich, H.-P. Waldl, H.G. Beyer, A. Luig, Short-term prediction of the aggregated power output of wind farms - a statistical analysis of the reduction of the prediction error by spatial smoothing effects, J. Wind Eng. Ind. Aerodyn. 90 (3) (2002) 231–246.
[19] J. Tastu, P. Pinson, E. Kotwa, H. Madsen, H.A. Nielsen, Spatio-temporal analysis and modeling of short-term wind power forecast errors, Wind Energy 14 (1) (2011) 43–60.
[20] ECMWF, Medium-range forecasts, 2016. Published online http://www.ecmwf.int/en/forecasts/documentation-and-support/medium-range-forecasts(Last visited: June 01, 2017).
[21] S. Späth, L. von Bremen, C. Junk, D. Heinemann, Time-consistent calibration of short-term regional wind power ensemble forecasts, Meteorol. Z. 24 (4) (2015) 381–392.
[22] EWEA, Wind in power: 2015 European statistics, 2016. Published online: https://windeurope.org/wp-content/uploads/files/about-wind/statistics/EWEA-Annual-Statistics-2015.pdf(Last visited: June 01, 2017).
[23] ECMWF, New forecast model cycle brings highest-ever resolution, 2016. Published online: http://www.ecmwf.int/en/about/media-centre/news/2016/new-forecast-model-cycle-brings-highest-ever-resolution (Last visited: June 01, 2017).
[24] C. Aggarwal, C. Reddy, Data clustering: Algorithms and applications, in: Chapman & Hall/CRC Data Mining and Knowledge Discovery Series, Taylor & Francis, 2013.
[25] Y.-J. Oyang, C.-Y. Chen, T.-W. Yang, A study on the hierarchical data clustering algorithm based on gravity theory, in: Proceedings of the 5th European Conference on Principles of Data Mining and Knowledge Discovery, PKDD '01, Springer Berlin Heidelberg, 2001, pp. 350–361.
[26] R.L. Thorndike, Who belongs in the family? Psychometrika 18 (4) (1953) 267–276.
[27] B.U. Schyska, A. Couto, L. von Bremen, A. Estanqueiro, D. Heinemann, Weather dependent estimation of continent-wide wind power generation based on spatio-temporal clustering, Adv. Sci. Res. 14 (2017) 131–138.
[28] R Core Team, R: A Language and Environment for Statistical Computing, R Foundation for Statistical Computing, Vienna, Austria, 2017. https://www.R-project.org/.
[29] B. Gräler, Developing Spatio-temporal Copulas (Ph.D. thesis), Westfälische Wilhelms-Universität Münster, 2014.
[30] T. Gneiting, F. Balabdaoui, A.E. Raftery, Probabilistic forecasts, calibration and sharpness, J. R. Stat. Soc. Ser. B Stat. Methodol. 69 (2) (2007) 243–268.
[31] T. Gneiting, A.E. Raftery, A.H. Westveld III, T. Goldman, Calibrated probabilistic forecasting using ensemble model output statistics and minimum CRPS estimation, Mon. Weather Rev. 133 (5) (2005) 1098–1118.
[32] A.E. Raftery, T. Gneiting, F. Balabdaoui, M. Polakowski, Using Bayesian model averaging to calibrate forecast ensembles, Mon. Weather Rev. 133 (5) (2005) 1155–1174.
[33] T. Jónsson, P. Pinson, H.A. Nielsen, H. Madsen, T.S. Nielsen, Forecasting electricity spot prices accounting for wind power predictions, IEEE Trans. Sust. Energy 4 (1) (2013) 210–218.
[34] T. Jónsson, P. Pinson, H. Madsen, H.A. Nielsen, Predictive densities for day-ahead electricity prices using time-adaptive quantile regression, Energies 7 (9) (2014) 5523–5547.

Marc Hüsch received his B.Sc. and his M.Sc. in Statistics from TU Dortmund University, Germany. During his studies, he was a visiting research intern at the National Renewable Energy Laboratory (NREL) in Golden (CO), USA and at the Center for Wind Energy Research (ForWind) in Oldenburg, Germany. He is currently pursuing a Ph.D. in Statistics at TU Dortmund University, where he is working on problems in the fields of data mining and spatial statistics.

Bruno U. Schyska received his B.Sc. in Physics from University Bremen, Germany, and his M.Sc. in Meteorology from Freie Universität Berlin, Germany. He is currently working as a research assistant in the energy meteorology group at University of Oldenburg's ForWind Center for Wind Energy Research in Oldenburg, Germany.

Lueder von Bremen received his Diploma in Meteorology at the University of Kiel, Germany in 1997 and his Ph.D. in 2001. From 2001 to 2005 he worked at the European Centre for Medium-Range Weather Forecasts and continued his career at ForWind, Center for Wind Energy Research at the University of Oldenburg, Germany. He has more than 15 years of experience in wind energy and Numerical Weather Prediction. Dr. von Bremen is interested in the design and management of the future European power supply system based on renewable energies and works in wind power forecasting using meteorological ensemble models. He joined the DLR-Institute of Networked Energy Systems (Oldenburg) beginning of 2018.